UNIVERSITÉ DE MONTRÉAL

CENTRE DE RECHERCHE SUR LES TRANSPORTS

CASE POSTALE 6128, SUCCURSALE A, MONTRÉAL, QUÉBEC, H3C 3J7

POSTMAN TOUR ON A GRAPH WITH PRECEDENCE RELATION ON ARCS

by

Moshe Dror*#, Pierre Trudeau**
and Helman Stern*

* Industrial Engineering and Management,
Ben-Gurion University of the Negev.
\# Visiting Professor, Centre de recherche sur les transports,
Université de Montréal, Summer 1984.

** Université de Montréal
Centre de recherche sur les transports.

Université de Montréal
Centre de recherche sur les transports - Publication #367
Septembre 1984

Dépôt légal - 4e trimestre 1984

ABSTRACT

Since the introduction of the Chinese Postman Problem - (CPP), many variations on the same theme have been developed. In this paper we examine still another variation. The arcs of the graph are partitioned and a precedence relation defined, specifying the order in which the elements of the partition have to be traversed. We first examine the conditions for a feasible solution to the problem. Next, we specify the graph properties of the precedence partition which insures a polynomial complexity solution of $O(N^5)$ where N is the number of nodes in the original graph.

I. INTRODUCTION

Given a connected graph G = (V,E) together with an additive nonnegative real (cost) function C on E - (the arcs of the graph G), the Chinese Postman Problem - (CPP) is defined as the problem of finding the least cost circuit on G which traverses at least once each element of E.

Many real life problems such as mail delivery, garbage collection, street cleaning, street inspection, road maintenance, snowplowing, home meter reading and others can be modeled as a variation of the CPP.

In this paper we address the case of a CPP with strict precedence relation on arcs. A strict precedence relation on arcs of G divides the elements of E into pairwise disjoint classes with certain priority index which indicates which class of arcs has to be traversed first, second, etc. A strict precedence means that no arc from, say, the second class can be traversed before the traversal of all the arcs from the first class is completed.

Before describing in detail the problem on hand and the proposed solution method we briefly review other work related to this research.

A concept closely related to the CPP is that of an Euler tour on a graph G. An Euler tour is a circuit which traverses each arc of a graph once and only once. If an Euler tour exists for a graph G, then it must be the optimal solution to the CPP on G. Ford and Fulkerson present in [5] (Theorem 7.1, p.60) the necessary and sufficient conditions for the existence of an Euler tour on a graph.

Theorem 1([5])[1]

The mixed 1-graph with no loops G = (V,E) contains an Euler tour if and only if; (a) G is connected; (b) every node of G is incident with an even number of arcs; (c) for every $X \subseteq V$, the difference between the number of directed arcs from X to $\bar{X}$ and the number of direted arcs from $\bar{X}$ to X is less than or equal the number of undirected arcs joining X and $\bar{X}$ ($\bar{X}$ = V-X).

In this paper we restrict ourselves to 1-graphs with no loops which are either totally directed or totally undirected. In the second case the Theorem 1 is reduced to conditions (a) and (b) only. The CPP on the type of graphs considered here is tractable (i.e., there exists polynomial time algorithms for this problem [9]). For the case of a general mixed graph the CPP is NP-hard (See [6], [10] for the complexity results).

Another version of the CPP which received attention in the literature is the capacitated CPP in which a given graph has to be covered in a cost effective manner by Euler paths (See [7]) or circuits each not exceeding a certain 'capacity'. Golden and Wong in [7] proved that even an approximate, restricted version of this problem is NP-hard.

A slightly different variation of the CPP is the Rural Postman Problem - (RPP) introduced by Orloff in [9]. This is the problem of finding the minimum cost tour which traverses each arc in a required subset of the arcs of a graph.

---

[1] Throughout this paper we use terms defined in Berge [2].

In [8] Leustra and Rinnooy Kan prove that the RPP is also an NP-hard problem. The added complexity of the RPP (versus the CPP) is contributed mainly to the number of the disconnected components of the required subset of arcs. As will shortly be seen, the CPP with precedence relation on arcs is threatened by the same danger.

II. Definition of Arc Precedence Relation on Graph

Let $G = (V,E)$ be a graph. Let $\{E_1,E_2,\ldots,E_k\}$ be a partition P_e of E such that for all $i \in E_i \subseteq E$, $E_i \cap E_j = \emptyset$ for $i \neq j$ and $\bigcup_{i=i}^{k} E_i = E$. A partition P_e together with an ordering relation $<$ defines an arc precedence relation in G in the following sence : for $E_i, E_j \in P_e$, $E_i < E_j$ means that in any feasible partial or complete traversal of arcs in E, arc $e \in E_i$ has to be traversed before any arc $f \in E_j$. We can always index the elements of P_e with respect to the order relation $<$ to obtain $E_1 < E_2 < \ldots < E_k$. We refer to the partition P_e with the relation $<$ as the precedence partition $(P_e, <)$ and to the graph G with $(P_e,<)$ as $(G,P_e,<)$(Though the same partition can have different ordering relations, with no loss to our analysis we assume one ordering relation to a graph).

A feasible postman tour cn $(G, P_e,<)$ is a tour which traverses each arc $e \in E$ at least once while preserving the order $<$ of the arcs. We distinguish between a closed postman tour in which the initial node is also the final node and an open postman tour in which the initial and final nodes do not have to coincide.

Naturally, the objective is to construct a feasible open (closed) postman tour of minimal lenght, on $(G,P_e,<)$ (Remark : throughout this paper the

terms cost, lenght and distance might be used interchangeably expressing the same cost measure of arc traversal).

III. Existence condition

We now present the existence conditions for the feasible postman tour. Some notation first.

Denote by G_1 the subgraph induced by the subset of arcs E_1 (i.e., all the arcs E_1 and incident nodes) and by G_2 the subgraph induced by $E_1 \cup E_2$. G_i is the subgraph induced by the subset of arcs $E_1 \cup E_2 \cup \ldots \cup E_i$ (clearly G is induced by $E_1 \cup \ldots \cup E_k$ if the graph does not have isolated nodes).

Theorem 2.1

Given an undirected graph G with precedence partition $(P_e,<)$ there exists a feasible open (closed) postman tour on $(G,P_e,<)$ if and only if (a) there exists an open (closed) postman tour on G; (b) there exists an open postman tour on G_i (with an end node incident to E_{i+1}) for $i = 1, \ldots, k-1$.

Proof : By induction on i.

In case of a directed graph we need to require more structure from G in order to establish the existence conditions for a feasible postman tour given a precedence partition $(P_e,<)$.

Theorem 2.2

Given a totally directed graph G with precedence partition $(P_e,<)$ there exists a feasible open (closed) postman tour on $(G,P_e,<)$ if

(a) there exists an open (closed) postman tour on G

(b) there exists an open postman tour on G_i for i=1, ..., k-1 (with the end node incident to E_{i+1})

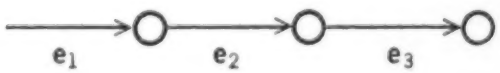
(c) the graph G_i for i=1, ..., k-1 is strongly connected.

The Theorem 2.2 contains only the sufficient conditions for the existence of a feasible postman tour in the case of totally directed graph. It is a weaker statement than that of Theorem 2.1

The additional requirement of directed traversal of arcs adds a degree of complexity to the postman problem given a precedence partition $(P_e,<)$.

The following example illustrates the role of part (c) in Theorem 2.2

Figure 1 A totally directed graph G.



Given the following precedence partition $(P_e,<)$: $e_2 \varepsilon E_1$, e_1, $e_3 \varepsilon E_2$ the conditions (a) and (b) are satisfied but there is no feasible postman tour on $(G,P_e,<)$.

Definition : For A_1, $A_2 \subseteq E$ the operation $A_1 \oplus A_2 = A$ is defined as that of duplicating the arcs in $A_1 \cap A_2$, i.e., the set A contains the set $A_1 \cup A_2$ and the set of arcs in $A_1 \cap A_2$ is duplicated in A.

The problem of finding a feasible postman tour on $(G,P_e,<)$ of minimal lenght is equivalent to the problem of finding a minimal lenght subset of arcs M such that the graph $G' \equiv (V, E \oplus M)$ satisfies the conditions of Theorem 2.1 in the undirected case and there exists a feasible postman tour in the totally directed case.

M is a subset of E in the sense that it can contain more than one identical element of E. Given the precedence partition $(P_e,<)$ we can view M as follows:

$M = M_1 \oplus M_2 \oplus \ldots \oplus M_k$ where

$M_1 \subseteq E_1$, $M_2 \subseteq (E_1 \oplus M_1) \cup E_2$, ..., $M_k \subseteq (E_1 \oplus M_1) \cup (E_2 \oplus M_2) \cup \ldots \cup E_k$. In order for G' to satisfy the conditions for the existence of a feasible postman tour the subgraphs G_1', G_2', ..., $G_k' = G'$ induced in turn by $E_1 \oplus M_1$, $(E_1 \oplus M_1) \cup (E_2 \oplus M_2)$, ..., $(E_1 \oplus M_1) \cup \ldots \cup (E_k \oplus M_k)$ have to contain an Euler path with appropriate end points for the undirected case and in the totally directed case. Denote by $\hat{E}_i$ the subgraph induced by the set of arcs E_i.

Note that G' is no longer a 1-graph with no loops but in general is a multigraph with no loops. We now assume that the original graph G is an undirected 1-graph with no loops and given a precedence partition $(P_e, <)$ the objective is to find a minimal lenght closed feasible postman tour with the starting node (the depot) given a priori (i.e., after completing the traversal of the arcs in E_k the postman has to return to the starting node incident to the arc subset E_1).

Next, for the undirected case, we outline an algorithm of polynomial time complexity which constructs a minimal lenght subset of arcs $M \subseteq V \times V$ such that the graph $G' = (V, E \oplus M)$ contains a feasible postman tour, given that $\hat{E}_i$ for i=1, ..., K are connected subgraphs, i.e., we construct an optimal postman tour.

The assumption that G_i is a connected graph for i=1,...,k is essential for the feasibility of the solution in respect to the precedence order imposed on the partition P_e. If the graph G_i is not connected but embeded in some larger graph say $\bar{G}_i$, which connects the components of G_i, then we have the case of the General Routing Problem which is NP-hard (See [8]).

We assume that such a subset of arcs M exists. We also assume that a node

designated as the depot and from which the tour originates and terminates is incident to E_1.

The Algorithm I

In this algorithm the main computational trust is directed towards building a (K + 1) - partite graph G" (See Figure 1) in which the node n_1^1 and the node n_1^{k+1} represent the depot. The nodes in the group N_2 represent all the nodes incident to the subset E_2 of arcs. The nodes in the group N_3 represent all the nodes incident to the subset E_3 of arcs, ...; the nodes in the N_k group represent all the nodes incident to the subset E_k of arcs. Note that a node in G can belong to more than one group N_i of nodes in G" if that node is incident to arcs of different subsets E_i.

Each node j in N_i denoted by n_j^i is connected by a directed arc with all the nodes in N_{i+1} (i=1,...,k). The cost of each such arc, say from the node n_j^i to the node n_ℓ^{i+1} represents the minimal cost of traversing all the arcs in the subset E_i possibly using arcs from $E_1,\ldots,E_{i-1}$ starting at the node n_j^i and ending at the node n_ℓ^{i+1} which is incident to the subset of arcs E_{i+1}. The calculation of this arc cost is of cube power time complexity in the number of nodes incident to the subset of arcs E_i. This time complexity corresponds to the complexity of the matching algorithm if required, to construct a minimal lenght path between the nodes n_j^i and n_ℓ^{i+1}.

A detailed description of the matching problem can be found in many references. We recommand specially [1] and [3]. Christofides in [3] provides an excellent general description of the different versions of the problem, outlines in some detail the solution algorithm and provides references for the computational behaviour of the solution algorithms. Ball and Dergis in [1] in

more recent paper analyze different strategies in coding a matching algorithm. Though slightly better bounds on the time complexity for the minimum weight matching algorithms are reported [3], we assume time complexity the cost of each arc of G" is of the order $O(N^3)$ where N represents the number of nodes in the graph G.

A rough upper bound on the time complexity which Algorithm I requires in order to construct the (k+1) - partite graph is obtained by calculating the upper bound on the number of arcs in the graph and multiplying by the time complexity for the matching algorithm.

The number of arcs required is clearly bounded by $(k-2)N^2 + 2N$ (where $|V|=N$). N). Thus, the time complexity of the graph construction algorithm is bounded by $(k-2)N^5 + 2N^4$. In order to find the minimal lenght feasible postman tour on our graph G what remains is to find a minimal cost unit flow from n_1 to n_1^{k+1} which is the shortest path between the two nodes n_1, n_1^{k+1}. The obvious result is that the time complexity of Algorithm I is bounded by $O(N^5)$.

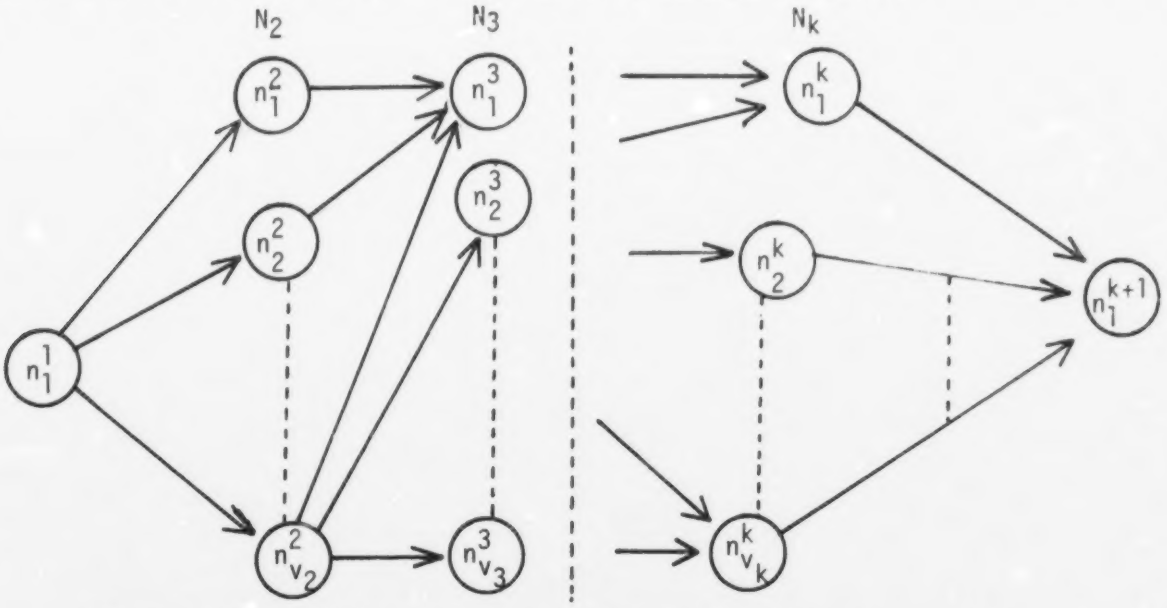
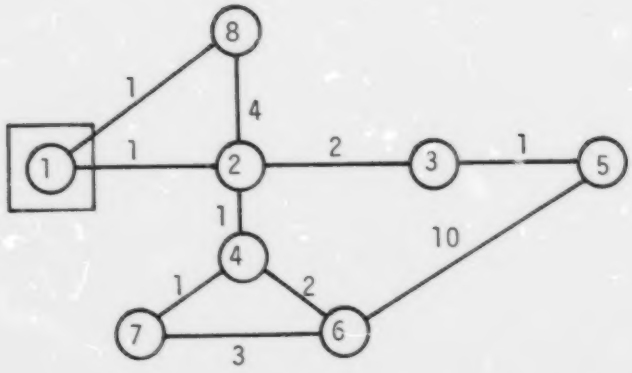


Figure 1 : The (k+1) - partite graph G". We denote by $v_i = |N_i|$.

Example



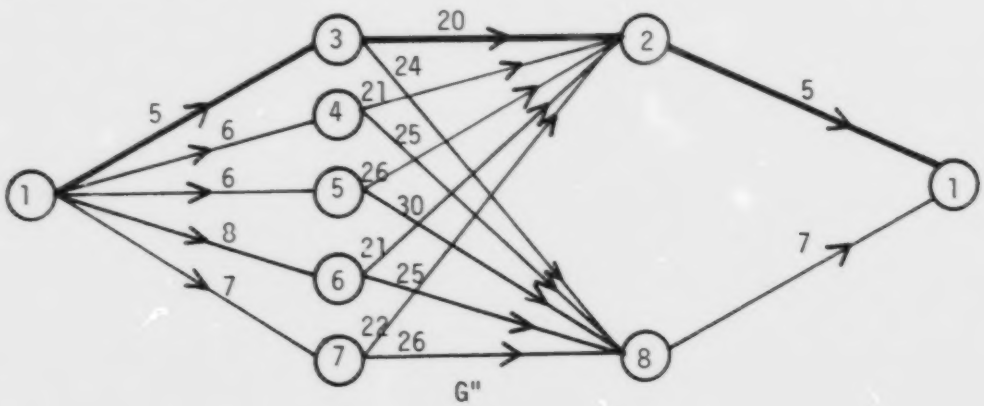
$P_e = \{E_1, E_2, E_3\}$

$E_1 = \{(1,2),(2,3),(2,4)\}$

$E_2 = \{(3,5),(5,6),(4,6),(4,7),(6,7)\}$

$E_3 = \{(2,8),(1,8)\}$

Where $E_1 < E_2 < E_3$ and the depot is node 1.
We search the minimal feasible closed postman tour on $(G, P_e, <)$.
G" is the following :



The shortest path in G" from 1 to 1 is ① → ③ → ② → ① and is of lenght 30. The solution in G is

① → ② → ④ → ② → ③ → ⑤ → ⑥ →
⑦ → ④ → ⑥ → ④ → ② → ⑧ → ①

At this point we examine our assumptions and ask whether they are essential in obtaining a polynomial time complexity for the procedure which constructs an optimal solution to the postman problem with precedence relation on arcs.

Given that a feasible solution exists, in the totally directed case, a similar algorithm can be applied in order to find the optimal postman tour. The change that will be required in this case is that of substituting a transportation problem solution for the matching problem solution (for Euler tour on totally directed graph see [2], [4], [6] or [10]). Also the assumption that G is a 1-graph with no loops can be dropped and multigraphs with loops can be considered. One has to adjust the subsequent matching problems (or transportation problems in the directed case) for that eventuality.

REFERENCES

[1] BALL, M., DERGIS, U., "An Analysis of Alternate Strategies for Implementing Matching Algorithms". Networks (1983), Vol. 13, pp. 517-549.

[2] BERGE, C., "Graphs and Hypergraphs". North Holland, Second, revised edition, reprinted 1979.

[3] CHRISTOFIDES, N., "Graph Theory, an Algorithmic Approach". Academic Press Inc. New York (1975).

[4] EDMONDS J. and JOHNSON E., "Matching, Euler Tours, and the Chinese Postman". Math. Prog.5, pp. 88-124 (1973).

[5] FORD, L.R., FULKERSON, D.R., "Flows in Networks". Princeton University Press, Sixth Printing, 1974.

[6] FREDRICKSON, "Approximation Algorithms for some Postman Problems". J. ACM.26, pp. 538-554 (1979).

[7] GOLDEN, B. and WONG, R., "Capacitated Arc Routing Problems". Networks, 11(3), pp. 305-315 (1981).

[8] LEUSTRA, J. and RINNOOY-KAN, A., "On General Routing Problems". Networks, 6(3), pp. 273-280 (1976).

[9] ORLOFF, C.S., "A Fundamental Problem in Vehicle Routing". Networks, 4, pp. 35-64 (1974).

[10] PAPADIMITRIOU, C.H., "On the Complexity of Edge Traversing". J. ACM, Vol. 23, No. 3, pp. 544-554 (1976).

[11] STERN, H. and DROR, M., "Routing Electric Meter Readers". Computers and Ops Res. 6, pp. 209-223 (1979).